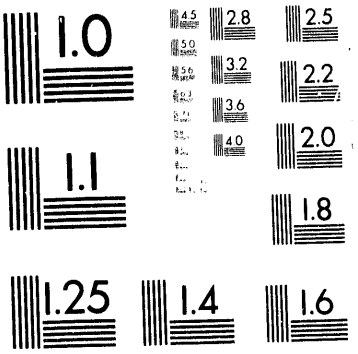
1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6

1 of 1

PNL-8282
UC-620

Integrated Baseline System (IBS)
Version 1.03

System Management Guide

January 1993

Prepared for the U.S. Department of Energy
under Contract DE-AC06-76RLO 1830

Pacific Northwest Laboratory
Operated for the U.S. Department of Energy
by Battelle Memorial Institute

DISCLAIMER

PACIFIC NORTHWEST LABORATORY
operated by
BATTELLE MEMORIAL INSTITUTE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC06-76RLO 1830

Printed in the United States of America

Available to DOE and DOE contractors from the Office of Scientific and Technical Information, P.O. Box 62, Oak Ridge, TN 37831; prices available from (615) 576-8401. FTS 626-8401.

Available to the public from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Rd., Springfield, VA 22161.

PNL-8282
UC-620

INTEGRATED BASELINE SYSTEM (IBS)
Version 1.03

SYSTEM MANAGEMENT GUIDE

J. R. Williams
S. Bailey[a]
J. C. Bower[b]

January 1993

Prepared for the
Federal Emergency Management Agency
under a Related Services Agreement
with the U.S. Department of Energy
Contract DE-AC06-76RLO 1830

Pacific Northwest Laboratory
Richland, Washington 99352

(a) Innovative Emergency Management
Baton Rouge, Louisiana

(b) Bower Software Services
Kennewick, Washington

NOTICE

The Integrated Baseline System (IBS) is an emergency planning and training tool intended to supplement existing resources for emergency planning, training, and management. IBS results should NOT be used as the sole basis for a decision regarding public safety. Although IBS is operational, not all IBS software/models have been verified. The assumptions of the IBS software should be reviewed to ensure its appropriateness for an intended use.

Preface

The Integrated Baseline System (IBS) is an emergency management planning and analysis tool being developed under the direction of the Federal Emergency Management Agency. The following documents were developed to support system users. The audience for each is identified.

This ***IBS User Guide*** explains how to start and use the ***IBS program***, which is designed to help civilian emergency management personnel to plan for and support their responses to a chemical-releasing ***event*** at a military chemical stockpile.[a] **Audience**: all users of the IBS, especially emergency management planners and analysts.

The ***IBS Utilities Guide*** explains how you can use IBS utility programs to manage and manipulate various kinds of IBS data. These programs include utilities for creating, editing, and displaying maps and other data that are referenced to geographic location. **Audience**: chiefly data managers but also system managers and some emergency management planners and analysts.

The ***IBS Models Guide*** summarizes the IBS use of several computer models for predicting the results of emergency situations. These include models for predicting dispersion/doses of airborne contaminants, traffic evacuation, explosion effects, heat radiation from a fire, and siren sound transmission. The guide references additional technical documentation on the models when such documentation is available from other sources. **Audience**: chiefly emergency management planners and analysts, but also data managers and system managers.

The ***IBS Data Management Guide*** provides the information needed to manage the data files and database used to support the administrative, user-environment, database management, and operational capabilities of the IBS. **Audience**: chiefly database administrators and system managers, but also emergency management planners and analysts who want to know details of the emergency management data.

The ***IBS System Management Guide*** defines IBS hardware and software requirements and gives instructions for installing, upgrading, or transferring the IBS software package. **Audience**: system managers.

(a) The IBS program was developed as part of the U.S. Army's Chemical Stockpile Emergency Preparedness Program (CSEPP).

About This Guide

Purpose

This *IBS System Management Guide* explains how to install or upgrade the Integrated Baseline System (IBS) software package. The IBS is an emergency management planning and analysis tool that was developed under the direction of the Federal Emergency Management Agency (FEMA).[a]

Scope

This guide includes detailed instructions for installing the IBS software package on a Digital Equipment Corporation (DEC) VAX computer from the IBS distribution tapes. The installation instructions include procedures for both first-time installations and upgrades to existing IBS installations.

To ensure that the system manager has the background necessary for successful installation of the IBS package, this guide also includes information on IBS computer requirements, software organization, and the generation of IBS distribution tapes.

When special utility programs are used during IBS installation and setups, this guide refers you to the *IBS Utilities Guide* for specific instructions. This guide also refers you to the *IBS Data Management Guide* for detailed descriptions of some IBS data files and structures.

Any special requirements for installation are not documented here but should be included in a set of ***installation notes*** that come with the distribution tapes.

Audience

This guide is addressed to the local system manager or administrator who is responsible for IBS system management—a person who has computer system management skills and authority, and who is familiar with both the DEC VMS operating system and the local VAX configuration on which the IBS is to be installed.

(a) The IBS is being developed by the Pacific Northwest Laboratory (PNL). PNL is operated for the U.S. Department of Energy by Battelle Memorial Institute under Contract DE-AC06-76RLO 1830.

Organization

This guide consists of the following five sections and four appendixes:

Section 1. Introduction

A description of the basic IBS software package and optional supplements.

Section 2. IBS Computer Requirements

A list of the hardware and software resources necessary and optional for operation of the IBS.

Section 3. Installation Procedures

Preparations and procedures for installing IBS for the first time or as an upgrade to an existing IBS installation. The first-time procedures include transferring the IBS from the distribution tapes, modifying system files and parameters, and managing user accounts.

Section 4. Software Organization

A summary of software organization: directory structures and file protections; command files for installation and start-up; and software standards for subdirectories, logical names, and symbols.

Section 5. How to Get Help

Names and addresses of people to contact about IBS and its installation.

Appendix A. Logical Names and Symbols

Lists of the logical names and symbols defined by IBS command files.

Appendix B. Generation of IBS Distribution Tapes

A procedure that explains how to prepare IBS distribution tapes.

Appendix C. Disk Requirements for Standard Topographic Overlays

Suggestions for estimating the storage requirements of 1:2,000,000-scale map data from the U.S. Geological Survey (USGS).

Appendix D. Rebuilding IBS from the Source Code

Instructions for rebuilding the entire IBS from its source code.

How to Use This Guide

This guide should be used in conjunction with whatever installation notes accompany the IBS distribution tapes.

For the First-Time Installer of IBS

If you are a first-time IBS installer, we recommend that you scan this document to get an idea of the scope of installation concerns. Then focus on *IBS Computer Requirements* (page 3) to verify that the system has adequate resources to install the software. Once adequate resources have been determined, proceed with *Installation Procedures* (page 5). Reference information about the IBS software setup is supplied in *Software Organization* (page 21).

For Experienced IBS System Managers

Experienced IBS installers and system managers should scan the document for areas of interest and then turn to *Installation Procedures* (if installation is of primary interest).

Visual Conventions

The following conventions are used in this guide.

Type Style	Meaning
bold	Text that you must type at the keyboard. Example: **$ NEWUSER**
italic	Place holders for information that you must provide. Example: **$ DEL** ***filename*** Here you would type the actual name of a file instead of the word shown in italics.
ALL CAPITALS	Directory names, file names, program names, and acronyms. Example: "Refer to the LOGIN.COM file in the [PNLDEV] directory."

Contents

Preface iii
About This Guide v
Tables xii

Section 1. Introduction 1

1.1 The Basic IBS Software Package 1
1.2 Supplements to the Basic Software Package 2
1.3 Getting Help 2

Section 2. IBS Computer Requirements 3

2.1 Hardware Requirements 3
2.2 Software Requirements 4

Section 3. Installation Procedures 5

3.1 Preparing for Installation of IBS 5
3.1.1 What to Do BEFORE Installing the IBS 5
3.1.2 Checking the IBS Distribution Tapes 6
3.2 Installing IBS for the First Time on a System 7
3.2.1 Transferring IBS from the Distribution Tapes 7
3.2.2 Modifying System Files and Parameters 9
3.2.3 Setting Directory and File Protections 13
3.2.4 Establishing New IBS User Accounts 15
3.2.5 Modifying Existing User Accounts for IBS Use 15
3.2.6 Creating an Operational Site 17
3.3 Installing IBS Software Upgrades 17
3.3.1 Upgrading to IBS 1.0*x* Map Databases 19
3.3.2 Upgrading to IBS 1.0*x* Site Databases 19

Section 4. Software Organization 21

4.1 Directories and Protections 21
4.2 IBS Command Files 24
4.3 IBS Software Standards 24
4.3.1 Subdirectories for Software Subsystems 25
4.3.2 Logical Names 25
4.3.3 Symbols 25

Section 5. How to Get Help 27

Appendix A: Logical Names and Symbols A.1

A.1 Disk Logical Names 2
A.2 IBS Logical Names and Symbols 2
A.3 *Map* Logical Names and Symbols 5
A.4 *Site* Logical Names 6
A.5 Other Logical Names and Symbols 8

Appendix B: Generation of IBS Distribution Tapes B.1

B.1 Preparing for an IBS Release 1
B.2 Preparing the Distribution Tapes 2

Appendix C: Disk Requirements for Standard Topographies C.1

Appendix D: Rebuilding IBS from the Source Code D.1

Tables

2.1 Minimal Vendor-Supplied Software for IBS 4
2.2 Additional Vendor-Supplied Resources for IBS 4

3.1 Save Set Files on the IBS Installation Tapes 7
3.2 Protections for IBS Directories 13
3.3 Protections for IBS Files 14
3.4 VMS Privileges 16
3.5 IBS Privileges 16
3.6 Process Quotas 16

4.1 IBS Directories and Protections 22
4.2 PNLDEV Subdirectories 23
4.3 Command Files that Support Installation/Startup 24

B.1 List of Save Sets and Procedures for Generating Tapes B.1

C.1 Worst-Case Block Size Estimates for IBS Topographic Data C.1

Section 1
Introduction

The Integrated Baseline System (IBS) was developed by the Pacific Northwest Laboratory (PNL) for the Federal Emergency Management Agency (FEMA). The intent of this document is to supply the information necessary for the reader to install and configure the IBS 1.0x software package on a Digital Equipment Corporation (DEC) VAX computer.

This section describes the basic IBS software package and optional supplements to the basic software package.

1.1 The Basic IBS Software Package

IBS consists of many software and database components. The "basic" IBS 1.0x software package includes software developed by PNL and a geographic database specific to each installation. This software package contains the basic functions that are expected to be part of most IBS installations throughout the emergency management community:

Function	IBS Component or Program
dispersion, deposition, decay, and dose model for radiological materials	MESORAD
chemical downwind hazard model	D2
chemical explosion-fire-dispersion models	CHEMS
dynamic traffic evacuation model	IDYNEV
regional evacuation analysis	REA
outdoor (siren) sound propagation model	OSPM
offpost emergency preparedness near a chemical stockpile	IBS
map display and editing of a geographic database	GIS
display of modeling results overlaid on maps	IBS, IEMIS

1.2 Supplements to the Basic Software Package

The following functions are NOT part of the basic IBS package and would be specific additions to particular FEMA installations:

- presentation graphics (DATAPLOT)
- Radiological Emergency Preparedness Support System (REPSS) or its supporting database
- expert systems model (TIMM—The Intelligent Machine Model)
- economic model (IMEASY)
- weather information access (WEATHER).

Software and databases beyond the basic IBS 1.0*x* software package must be acquired from their originating sources.

1.3 Getting Help

For details on other capabilities or for assistance beyond the instructions provided here, contact the FEMA IBS system manager. See *How to Get Help*, page 27, for addresses and telephone numbers.

Section 2
IBS Computer Requirements

The IBS has specific computer hardware and software requirements. This section lists the basic hardware requirements as well as the requirements for vendor-supplied software.

2.1 Hardware Requirements

The IBS software requires as a *minimum* the following computer hardware:

- Digital Equipment Corporation (DEC) VAX computer system that is capable of running VMS (examples include the VAX 4000 and MicroVAX II)
- 32 megabytes (MB) of random access memory (RAM) **plus** 1 MB per user
- 1.5 to 4.5 gigabytes (GB) of mass storage
- Tektronix 4107 terminals, Tektronix 4107 PC terminal emulators, or an equivalent
- asynchronous interface (ports for Tektronix terminals)
- magnetic tape storage unit (TK50 or TK70 tape drives)
- high-density backup device (4-mm or 8-mm)
- a modem capable of operating at 9600 baud or faster.

This minimum hardware configuration provides adequate resources for a single user to operate IBS on test data sets and small local area databases and allows the user to view the results on a computer display. Higher performance VAX computers may be desired for multiple-user environments. Generally, each concurrent user of IBS should have 1 MB of memory, one asynchronous port (unless part of a network), and 1 million instructions per second (MIPS) of central processing unit (CPU) computing power.

Desirable Additions. Desirable capabilities beyond the basic hardware requirements include the following:

- more on-line mass storage (several GB may be necessary)

- a hard copy device (such as a line printer) or a color hard copy device (for the Tektronix terminal).

These listed hardware requirements represent a simplification of the actual requirements for IBS. They are intended to be sufficient information for discussing IBS requirements with a VAX system manager or a computer vendor. Please consult Appendix C for more information on the storage requirements of map databases. This is a significant factor in sizing mass storage devices.

2.2 Software Requirements

The PNL-developed software and basic geographic database are included in the IBS software package supplied by FEMA. As a minimum, the basic operation of IBS also requires the vendor-supplied software shown in Table 2.1.

For the IBS installation at FEMA headquarters and for other installations requiring IBS capabilities beyond the basic IBS functions, the vendor-supplied resources listed in Table 2.2 may be required.

Table 2.1. Minimal Vendor-Supplied Software for IBS

Software	Vendor	Version
VAX/VMS Operating System (or MicroVMS for the MicroVAX)	DEC	5.4

Table 2.2. Additional Vendor-Supplied Resources for IBS

Software	Vendor	Version	Function
DECnet	DEC	Same as VAX/VMS	Network with IBS facilities
FORTRAN	DEC	5.7(1)	Modify IBS source code
TIMM(2)	*See the IBS System Manager for specific information.*		Generate "expert system" applications
Weather Access	*See the IBS System Manager for specific information.*		Obtain periodic weather data
DATAPLOT	*See the IBS System Manager for specific information.*		Presentation graphics

(1) The version numbers specified are those currently set up on the PNL IBS VAX in Richland, Washington. Newer releases of this software may exist. A qualified computer professional should review the installation of versions different from those specified in the above chart.
(2) The Intelligent Machine Model (TIMM)

Section 3
Installation Procedures

Procedures for installing the IBS software package are documented in this section. This discussion includes 1) Preparing for Installation of IBS, 2) Installing IBS for the First Time on a System, and 3) Installing IBS Software Upgrades. (This discussion assumes that the DEC VMS operating system is installed and operating on the computer. If VMS is not installed, have it installed according to the instructions provided by the vendor.)

3.1 Preparing for Installation of IBS

To prepare for installing the IBS software package, please heed the following instructions and review the description of the IBS distribution tapes.

3.1.1 What to Do BEFORE Installing the IBS

The following preparatory steps are **strongly recommended**:

1. Before installing the IBS Software Package provided, please review *IBS Computer Requirements* (page 3) to verify that your computer has adequate resources for the intended implementation.

   Evaluation of available computer resources is especially important when installing IBS on computers that are running other software and have multiple uses. When the available computer resources are inadequate or "just" adequate for the intended IBS implementation, action can be taken to alleviate any deficiencies before the installation. An example of a potential deficiency is inadequate disk space for the software and data.

2. If you are not also the computer system manager (person responsible for the computer system) at your local VAX installation, then plan and coordinate IBS installation with that system manager.

3. To protect against the inadvertent permanent loss of old information contained on the computer, **perform a complete system backup** before installing the IBS software or upgrade.

4. Always refer to any installation notes that may accompany the distribution tapes.

3.1.2 Checking the IBS Distribution Tapes

IBS is distributed on TK50 or TK70 tapes in VMS Backup format. Each set of installation tapes should be accompanied by installation notes or instructions that supplement the instructions in this *System Management Guide*. Seven backup save sets will be included on the distribution tapes:

- ISYSFILES.BCK contains system data files.
- PNLDEV.BCK contains all source code, build files, and executables.
- OFFSITE.BCK contains IBS system-wide files. It should be mostly empty, containing at the very least the site subdirectory corresponding to the one in INFOMANAGER together with the Emergency Functions file.
- ONSITE.BCK contains files for the communication interface between IBS and the onpost system. It should be empty, except for JUNK.TMP.
- INFOMANAGER.BCK contains the operational site database and other site files set up as they would be for an IBS user.
- TRNG.BCK contains text and materials about IBS training.
- ***Map***.BCK contains a regional geographic (map) database. (For example, TEADDB.BCK contains a map database for Tooele, Utah. The database provided may vary, depending on the installation. Refer to any special installation instructions that come with the distribution tapes.)

Each of the backup save sets, ISYSFILES.BCK, PNLDEV.BCK, OFFSITE.BCK, ONSITE.BCK, INFOMANAGER.BCK, TRNG.BCK, and ***Map***.BCK, correspond to the principal IBS directories: [ISYSFILES], [PNLDEV], [OFFSITE], [ONSITE], [INFOMANAGER], [TRNG], and [***Map***]. The IBS installation procedures will use (or create) these directories—see *Software Organization*, page 21.

As shown in Table 3.1, all files are required for the initial installation of IBS on a specific computer. For an upgrade of the IBS software on an existing IBS installation, only the PNLDEV.BCK save set may be required: check the installation instructions received with the tape.

Table 3.1. Save Set Files on the IBS Installation Tapes

Save Sets Required for IBS Installations	
FIRST Installation	ALL Installations
PNLDEV.BCK	PNLDEV.BCK
ISYSFILES.BCK(1)	--
OFFSITE.BCK	--
ONSITE.BCK	--
INFOMANAGER.BCK	--
Map.BCK	--
TRNG.BCK	TRNG.BCK

(1) May be required on some installations other than the first.

3.2 Installing IBS for the First Time on a System

The following sections explain how to install the IBS 1.0x software package for the first time[a] on a system. The instructions include these steps:

1. Transferring IBS from the Distribution Tapes
2. Modifying System Files and Parameters
3. Setting Directory and File Protections
4. Establishing New IBS User Accounts
5. Modifying Existing User Accounts for IBS Use
6. Creating an Operational Site.

3.2.1 Transferring IBS from the Distribution Tapes

Note: In the following instructions, the italicized *device names* are place holders that represent the logical names that you may have already assigned to one or more physical disk drives or tape drives on your system. In procedures, substitute the correct logical names of your devices for the italicized place-holder names. Note that the four logical disk drives in the following list could all be the same physical drive if one drive has the capacity to hold all the information.

- *DiskA* represents a logical name assigned to the disk drive where the source code and system data files will reside. (The [PNLDEV] and [ISYSFILES]

(a) "First time" installation means that no versions of IEMIS/SPOCC/IBS have been previously installed on the system. IEMIS is FEMA's Integrated Emergency Management Information System. SPOCC was the precursor to the IBS program for chemical stockpile emergency preparedness.

directories must reside on this disk unit.) *DiskA* will be PNLDEVDB on systems with existing IBS installations.

- *DiskB* represents a logical name assigned to the disk drive where the INFOMANAGER user account and the modifiable parts of the IBS operational site databases will be found. The other files related to IBS (from the OFFSITE and ONSITE save sets) also will be found there. *DiskB* will be EVNDB on systems with existing IBS installations.

- *DiskC* represents a logical name assigned to the disk drive where training text files will be found. *DiskC* will be TRNGDB on systems with existing IBS installations.

- *DiskD* represents a logical name assigned the disk drive where the map database will be found. *DiskD* will be *Map*DB on systems with existing IBS installations.

- *TapeA* represents a logical name assigned to the TK50 or TK70 tape drive.

Use the following series of commands in sequence, substituting the correct logical names for the italicized place holders where appropriate. Notice that the command procedures are the same for each set of data or software: only the disk device is different.

1. Give yourself the necessary VMS privilege:
   $ SET PROC/PRIV=SYSPRV

2. Restore ISYSFILES.BCK and PNLDEV.BCK:
   $ MOUNT/FOREIGN *TapeA*:
   $ BACKUP/NEW/LOG *TapeA*:*.* *DiskA*:[*...]
   $ DISMOUNT *TapeA*

3. Restore OFFSITE.BCK, ONSITE.BCK, and INFOMANAGER.BCK:
   $ MOUNT/FOREIGN *TapeA*:
   $ BACKUP/NEW/LOG *TapeA*:*.* *DiskB*:[*...]
   $ DISMOUNT *TapeA*

4. Restore TRNG.BCK:
   $ MOUNT/FOREIGN *TapeA*:
   $ BACKUP/NEW/LOG *TapeA*:*.* *DiskC*:[*...]
   $ DISMOUNT *TapeA*

5. Restore *Map*.BCK:
   $ MOUNT/FOREIGN *TapeA*:
   $ BACKUP/NEW/LOG *TapeA*:*.* *DiskD*:[*...]
   $ DISMOUNT *TapeA*

The IBS software and supplied data have now been transferred to the computer. The next step is to modify the VMS and IBS software to set up IBS for use.

3.2.2 Modifying System Files and Parameters

Installation of IBS on a new computer requires changes to some of the VMS system files so that the IBS setup command files are invoked. Then some of the IBS startup files must be modified to reference devices and software peculiar to that system. Next, some system parameters will probably need to be changed, among them the size of the detached process logical name table and the virtual page count limit.[a] Finally, the system must be initialized for IBS by invoking the necessary command files or by rebooting the system.

3.2.2.1 Modifying VMS System Files

1. SYS$MANAGER:SYSTARTUP_V5.COM

   Add the following lines at the bottom of the file:
   $ ASSIGN/SYSTEM *DiskA*: PNLDEVDB
   $ @PNLDEVDB:[ISYSFILES]PNLLOGDEF
   $ @PNLDEVDB:[ISYSFILES]PNLEXECUT

   Substitute the correct logical device name for the italicized name shown here. (*DiskA* is the drive that contains the IBS software.)

2. SYS$MANAGER:SYLOGIN.COM

   Add the following line at the bottom of the file:
   $ @PNLDEVDB:[ISYSFILES]PNLSYMDEF

3.2.2.2 Modifying IBS System Files

1. *DiskA*:[ISYSFILES]ASS_DISK_LOG.COM

   Modify the logical device names:
   $ ASSIGN/NOLOG/SYSTEM *DiskB*: EVNDB/EXEC
   $ ASSIGN/NOLOG/SYSTEM *DiskC*: TRNGDB
   $ ASSIGN/NOLOG/SYSTEM *DiskD*: *Map*DB

(a) It has been reported that other system parameters may need to be changed also. These include GBLSECTIONS (450), GBLPAGFIL (1024), and GBLPAGES (62500). While IBS itself does not require these changes, when used in conjunction with other software on the system, IBS may not run without making the modifications.

Substitute the correct logical device names for the italicized names shown here. (Again, *DiskB* contains the IBS event database; *DiskC* contains training materials; *DiskD* contains the map database.)

2. *DiskA*:[ISYSFILES]ASS_SITE_LOG.COM[a]

   Substitute the correct values for the following symbols:
   $ TDISK := EVNDB:[b]
   $ TUSER := INFOMANAGER[c]
   $ TSITE := *Site*

   TSITE is used to define the EVNT$DISK logical, the home disk for the Information Manager user. TUSER is used to define the EVNT$USER logical, the username for the Information Manager. TSITE is used to define the EVNT$SITE logical, the default operational site.

3. *DiskA*:[ISYSFILES]ASS_VENDOR_SYM.COM

   If you have the Presentation Graphics software package on your system, then edit the file and ensure that the DP symbol runs the software.

   If you have the Weather software package on your system, then edit the file and ensure that the WEATHER symbol runs the weather software.

4. *DiskA*:[ISYSFILES]ASS_VENDOR_LOG.COM

   If you have the TIMM (The Intelligent Machine Model) software package on your system, then edit the file and ensure that the SYS$TIMM logical points to the TIMM software.

5. *DiskA*:[ISYSFILES]OTHER_SYSTEMS.DAT

   Edit this file and include the names of only those offpost systems to which the EVENT program should route onpost information. For more information, refer to OTHER_SYSTEMS.DAT in the *IBS Data Management Guide*, Appendix A. An example follows:

   ```
   KIRK"INFOMANAGER INFOMGR"::EVNDB:[ONSITE]
   PICARD"INFOMANAGER FLEZPUTR":EVNDB:[ONSITE]
   ```

   After saving the file, set the protections so that no one else can look at it:

(a) For new installations of IBS, this file should be correct.

(b) The default is EVNDB: and it should not be necessary to change it.

(c) The default is INFOMANAGER and it should not be necessary to change it.

```
$ SET PROT=(S:RWED,O:RWED,G,W) DiskA:[ISYSFILES]OTHER_SYSTEMS.DAT
```

6. You may want to edit the following files and configure them for your system. Refer to the *IBS Data Management Guide* for the structure and content of each file. All these files are located in *Disk A*:[ISYSFILES]:

   - LOGIN.COM_TEMPLATE
   - JOB_ENV.DAT_TEMPLATE
   - SITE_EF.DAT_TEMPLATE
   - TOPODESC.FIL_TEMPLATE.

3.2.2.3 Modifying System Parameters

The size of the system's detached process logical name table MUST be adjusted to match the size of the job logical name table authorized for an IBS user. This involves setting the PQL_MJTQUOTA and PQL_DJTQUOTA parameters to 0, which causes the user's JTQUOTA value[a] in the User Authorization File (UAF) to be used for each detached process. To modify these parameters, run SYSGEN as shown in the following sequence of commands:

	Command	Result / Comments
1.	**$ SET DEF SYS$SYSTEM:**	
2.	**$ MCR SYSGEN**	Runs SYSGEN
3.	SYSGEN> **USE CURRENT**	Uses current system parameter file
4.	SYSGEN> **SET PQL_MJTQUOTA 0** SYSGEN> **SET PQL_DJTQUOTA 0**	Sets current values to 0
5.	SYSGEN> **WRITE ACTIVE**	Writes the new values to the active parameters in memory
6.	SYSGEN> **WRITE CURRENT**	Saves the current parameter values to the system parameter file

If the system virtual page count limit (VIRTUALPAGECNT) is less than about 200,000, you will need to change this parameter also. To show and then modify the current limit, run SYSGEN as shown in the following sequence of commands.

(a) The NEWUSER utility sets JTQUOTA to 8192.

	Command	Result / Comments
1.	**$ SET DEF SYS$SYSTEM:**	
2.	**$ MCR SYSGEN**	Runs SYSGEN
3.	SYSGEN> **USE CURRENT**	Uses current system parameter file
4.	SYSGEN> **SHOW VIRTUALPAGECNT**	Shows the current parameter value
5.	SYSGEN> **SET VIRTUALPAGECNT 200000**	Modifies VIRTUALPAGECNT to be 200,000
6.	SYSGEN> **SHOW VIRTUALPAGECNT**	Verifies the new value of 200,000
7.	SYSGEN> **WRITE CURRENT**	Saves the current parameter values to the system parameter file
8.	SYSGEN> **SHOW VIRTUALPAGECNT**	Verifies the saved parameter value
9.	SYSGEN> **CREATE SYS$SYSTEM:PAGEFILE1/SIZE=200000/CONTIG**	Creates a contiguous PAGEFILE of at least 200,000 blocks
10.	SYSGEN> **INSTALL SYS$SYSTEM:PAGEFILE1.SYS/PAGEFILE**	Installs the new PAGEFILE
11.	SYSGEN> **EXIT**	Exits SYSGEN
12.	Edit SYS$SYSTEM:MODPARAMS.DAT and change the value for VIRTUALPAGECNT to 200000.	*Note: MODPARAMS.DAT may not exist on all systems. If it does not exist, create it and include the following line:* VIRTUALPAGECNT=200000

3.2.2.4 Initializing the System for IBS

After modifying system parameters (as explained in the previous section), carry out the following steps for initializing the system for IBS.

1. Reboot the system[a], OR enter the following commands:
   $ ASSIGN/SYSTEM *DiskA*: PNLDEVDB
   $ @PNLDEVDB:[ISYSFILES]PNLLOGDEF
   $ @PNLDEVDB:[ISYSFILES]PNLSYMDEF
   $ @PNLDEVDB:[ISYSFILES]PNLEXECUT

   Note: Table 4.3 describes the three command files used here. If things have not been correctly set up, the PNLEXECUT command file can fail to start the EVENT and Mail Checker batch processes for IBS. If that happens, you should see error messages while PNLEXECUT is executing.

(a) You must reboot the system if you changed the VIRTUALPAGECNT because it is not a dynamic parameter.

2. To check whether the EVENT and Mail Checker processes started, use either the WHO or SHOW SYSTEM commands and see if "EVENT - BATCH" and "Mail Checker" are executing.

   If EVENT is not executing, examine the EVENT.LOG file in the INFOMANAGER directory to determine the problem. Among the possible problems that would prevent these processes from executing is that 1) the INFOMANAGER account is not set up (by using NEWUSER) or that 2) no site has been created for the INFOMANAGER account (by using NEWSITE). The NEWUSER utility may be needed to recreate the INFOMANAGER account, even when the system is upgraded from an earlier version of the system.

3. Add the MYTAPE, CHAT, and WHO verbs to the DCLTABLES:
   $ @IEMIS$EXE:ADDVERB

4. Use the DELPTDB utility to initialize data values in the people tracking database used by the LOCATE program:
   $ DELPTDB

3.2.3 Setting Directory and File Protections

Use the SET PROT command to set the correct protections on the IBS directories listed in Table 3.2. (Section 4 identifies the contents of these IBS directories.)

EXAMPLE: The following command sets protections for the IBS software directory [PNLDEV]:

$ SET PROT=(S:RWED,O:RWED,G:RE,W:RE) [000000]PNLDEV.DIR

Table 3.2. Protections for IBS Directories

$ SET PROT=	for these IBS directories
(S:RWED,O:RWED,G:RE,W:RE)	[000000]PNLDEV.DIR
	[000000]*Map*.DIR
	[000000]TRNG.DIR
(S:RWED,O:RWED,G:RWE,W:RWE)	*Map*DB:[*map*]LOG.DIR
(S:RWED,O:RWED,G:RWED,W:RWED)	[000000]ISYSFILES.DIR
	[000000]INFOMANAGER.DIR
	[000000]OFFSITE.DIR
	[000000]ONSITE.DIR

Then set the correct protections on the various IBS files or groups of files listed in Table 3.3. Notice that some protections must be set in the specific order shown in the table so that specific files have protections that vary from a group of files.

EXAMPLE: The first command below sets protections for all files in the IBS system directory [ISYSFILES]. The second command then resets different protections for a specific file in that directory, ALLSITE.DAT:

```
$ SET PROT=(S:RWED,O:RWED,G:RE,W:RE)  [ISYSFILES]*.*
$ SET PROT=(S:RWED,O:RWED,G:RWED,W:RWED)  [ISYSFILES]ALLSITE.DAT
```

Table 3.3. Protections for IBS Files

$ SET PROT=	for these IBS files
(S:RWED,O:RWED,G:RE,W:RE)	[PNLDEV...]*.*
	[ISYSFILES]*.*
	[*Map*...]*.*
	[TRNG...]*.*
(S:RWED,O:RWED,G:RWE,W:RWE)	[ONSITE]*.*
	[OFFSITE...]*.*
	[INFOMANAGER...]*.*
(S:RWED,O:RWED,G:RWED,W:RWED)	[ISYSFILES]ALLSITE.DAT
	[ISYSFILES]MASTER_DIRECTORY.BIN
	[ISYSFILES]OFF_TO_ON.DAT
	[ISYSFILES]PERMITI.DAT
(S:RWED,O:RWED,G,W)	[ISYSFILES]OTHER_SYSTEMS.DAT
(S:RWED,O:RWED,G:RWE,W:RWE)	[PNLDEV.EXE]SPOCC_SHARED_MEMORY.EXE
	MAP$MEMLOG:MEM_LOG.DAT (1)

(1) If MEM_LOG.DAT does not exist initially, the IBS software will try to create it.

These IBS directory and file protections are also summarized in a single table (4.1) in Section 4.

3.2.4 Establishing New IBS User Accounts

Use the NEWUSER utility to create IBS user accounts. You must have SYSPRV privilege to run this utility. After you supply the user name, full user name, disk, account, group, and member[a] information, NEWUSER creates a command file (NEWUSER.*username*) in your login directory, which it then executes to create the user account in the UAF. NEWUSER then

- creates the necessary directories for the user
- copies various files from IEMIS$SYSFILES: to the home directory
- sets file and directory ownership and protections
- adds a record to USERNAME.DAT, the master list of IBS users who can receive electronic mail through IBS
- adds a permission record (with limited privileges) to IEMIS$SYSFILES:PERMITI.DAT (the IBS authorization file).

Refer to the *IBS Utilities Guide* for specific information about using NEWUSER.

Changing User Privileges. If a new user needs to have different IBS privileges than those granted by the NEWUSER utility, use the PDMGR utility (Permiti.Dat Manager) to change privileges. Refer to the *IBS Utilities Guide* for information about using PDMGR.

Updating the List of Electronic Mail Receivers. Although the NEWUSER utility adds the user to the IBS master list of electronic mail receivers, you must use the MLMGR (Mail Manager) utility to complete the new user's mail record by adding job title information. Refer to the *IBS Utilities Guide* for information about using MLMGR.

3.2.5 Modifying Existing User Accounts for IBS Use

Existing user accounts may not have the correct directories, the correct privileges, or the correct process quotas to use IBS. The NEWUSER utility should also be used to update existing user accounts, giving them the directories, privileges, and quotas they require.

(a) The group and member are octal numbers used for the User Information Code (UIC). The group numbers should be the same for all IBS users, while the member numbers MUST be unique. The system manager should look at the UAF to find an unused group and put all IBS users in it. Refer to VAX/VMS documentation for more information about the UAF file.

The privileges granted IBS users by the NEWUSER utility are shown in Tables 3.4 and 3.5[a].

The process quotas set by the NEWUSER utility are shown in Table 3.6.

Table 3.4. VMS Privileges

TMPMBX
NETMBX
EXQUOTA
SYSLCK
GRPPRV

Table 3.5. IBS Privileges

EESF
REA
Map Display
Graphics Editor
Map Utilities
Training Menu
CHEMS
OSPM
MESORAD
Create New Site
GEDIT
Display Site Map
Display Evac Results
Print Reports
Copy from
IEMIS User
IBS User
General User

Table 3.6. Process Quotas

Quota Parameter	Value
ASTLM	24
BIOLM	24
BYTLM	65536
CPUTIME	0
DIOLM	18
ENQLM	600
FILLM	70
JTQUOTA	8192
MAXACCTJOBS	0
MAXJOBS	0
PBYTLM	0
PGFLQUOTA	200000
PRCLM	4
PRIORITY	4
PWDMINIMUM	8
QUEPRIO	0
SHRFILLM	0
TQELM	10
WSDEFAULT	150
WSEXTENT	4096
WSQUOTA	1024

Changing User Privileges. If a new user needs to have different IBS privileges than those granted by the NEWUSER utility, use the PDMGR utility (Permiti.Dat Manager) to change privileges. Refer to the *IBS Utilities Guide* for information about using PDMGR.

(a) The Information Manager user is granted ALL privileges, though SYSPRV is the only additional privilege set by default.

3.2.6 Creating an Operational Site

By default, the initial installation of IBS should include an operational site (a set of operational site data) that is already set up for IBS use. Refer to the installation notes that accompany the distribution tapes: if the installation does NOT include an operational site, then you (the system manager) or a database administrator must use the NEWSITE utility to create a new operational site. Refer to the *IBS Utilities Guide* for information about using NEWSITE.

3.3 Installing IBS Software Upgrades

The following software upgrade instructions are provided for systems that are upgrading the IBS software but retaining the existing databases on the system. If you are not familiar with the IBS logical name assignments, see *Logical Names* (page 25) and *Modifying System Files and Parameters* (page 9) before following these instructions.

1. Back up your disk drives before starting the installation of the IBS upgrade.

2. Make sure no one is using the IBS software.

3. Give yourself the necessary VMS privileges:
   $ SET PROCESS/PRIVILEGE=(SYSPRV,BYPASS)

4. Move to the appropriate disk:
   $ SET DEFAULT PNLDEVDB:[000000]

5. If the upgrade includes the [ISYSFILES] directory (refer to the installation notes), then save the current directory by renaming it:
   $ RENAME ISYSFILES.DIR ISYSFILES_OLD.DIR

6. De-install the shared image file:
   $ @PNLDEVDB:[PNLDEV.RELEASE]INSTALL-OFF

   If you do not do this, you will NOT be able to delete the directory in the next step.

7. Delete the current IBS software:
   $ DELETE [PNLDEV...]*.*;*

   Repeat this command until all files and subdirectories have been deleted. This process can take a considerable amount of time (more than 30 minutes is not unreasonable).

8. Load the IBS software upgrade.

   First, mount the tape:
   $ MOUNT/FOREIGN *TapeA*:

   Now load the actual IBS software:
   $ BACKUP/LOG/NEW *TapeA*:*.* [*...]

   This should take 2–4 hours. The software may be on more than one tape, so you will need to remove the first tape and insert the second when prompted to do so.

   Next, dismount the tape:
   $ DISMOUNT *TapeA*:

9. If the upgrade includes the [ISYSFILES] directory, then restore selected files from the [ISYSFILES_OLD] directory. Refer to the installation notes for the exact list. The commands will be something like the following:

   $ COPY [ISYSFILES_OLD]*xx* [ISYSFILES]*.*

   When done restoring the old files, delete the old directory:
   $ @IEMIS$EXE:DELTREE [ISYSFILES_OLD]

   If you did not restore the old version of the OTHER_SYSTEMS.DAT file, then set the protections on it so that no one can read it:
   $ SET PROT=(S:RWED,O:RWED,G,W) [ISYSFILES]OTHER_SYSTEMS.DAT;*

10. If the directory and file protections are incorrect, set them to the correct values for the directories and files included in the upgrade. Refer to *Setting Directory and File Protections*, page 13.

    At this stage, the IBS software upgrade has been installed. For IBS to be fully operational, you may now need to modify some system files, system parameters, and user accounts. Please refer to

    - *Modifying System Files and Parameters*, pages 9 through 15.
    - *Modifying Existing User Accounts for IBS Use*, page 15.

11. At the very least, you must initialize the system for the upgraded IBS and define the logical names and symbols used to install the images by executing the following command files and utility program (see *Initializing the System for IBS*, page 12.):

```
$ @[ISYSFILES]PNLLOGDEF
$ @[ISYSFILES]PNLSYMDEF
$ @[ISYSFILES]PNLEXECUT
$ DELPTDB
```

Table 4.3 describes the three command files used here. The DELPTDB utility program clears old data and resets startup values in the current "people tracking database" used by the LOCATE program. This is normal procedure unless you want to retain the current values of the people tracking database for some reason.

3.3.1 Upgrading to IBS 1.0x Map Databases

If the map databases on your system do not conform to specifications for IBS version 1.0 or greater, you will need to convert them to the new structure using the FIXMAPDB utility. It creates a SETUPDB.COM file, a TOPOCTL.FIL, and a TOPODESC.FIL. After the execution of FIXMAPDB is complete, the map database is available to IBS users. Please refer to the *IBS Utilities Guide* for information on FIXMAPDB and to the *IBS Data Management Guide* for IBS 1.0 map database specifications.

3.3.2 Upgrading to IBS 1.0x Site Databases

If the user site databases on your system do not conform to IBS 1.0 specifications or greater, all users must convert their own sites to the new structure using the FIXSITEDB utility. FIXSITEDB keeps model data, population data, and static topography data but throws out volatile topography data, site control files (SETUP.COM is recreated and moved; CASEINDEX.DAT is converted) and any data in the [*user*.SITES*site*.INPUT.EMP] or [*user*.SITES*site*.INPUT.OFF] directories. A [*user*.SITES*site*.INPUT.IP] directory is created if necessary. Refer to the *IBS Utilities Guide* for information on FIXSITEDB and to the *IBS Data Management Guide* for IBS 1.0 site database specifications.

Section 4
Software Organization

This section presents a summary of the software organization. This summary includes the high-level directory structures; command files for installation and start-up; and software standards for subdirectories, logical names, and symbols.

4.1 Directories and Protections

Seven directories are required for IBS.[a]

- **PNLDEVDB:[PNLDEV]** -- This is the top-level directory for the IBS software package and contains all documentation, source code, executables, and command files necessary for building and executing the software developed by PNL. (No third-party software is contained in this directory.)

- **PNLDEVDB:[ISYSFILES]** -- This directory contains template files, onpost simulator input file, and site-dependent control files, all maintained by the database administrator.

- **EVNDB:[ONSITE]** -- This directory contains onpost/offpost interface files.

- **EVNDB:[OFFSITE]** -- This is the top-level directory for shared IBS site files.

- **EVNDB:[INFOMANAGER]** -- This is the home directory for the Information Manager user AND the top-level directory for the operational site databases.

- **TRNGDB:[TRNG]** -- This directory contains IBS training materials.

- ***Map*DB:[*Map*]** -- This is the top-level directory of the map database. For example, the top-level directory of the Tooele map database is TEADDB:[TEAD].

Table 4.1 shows the protections assigned to these IBS directories and the files within them.

The [PNLDEV] directory contains subdirectories for each IBS subsystem. Table 4.2 outlines the first level of subdirectories in [PNLDEV].

(a) For more information on the structure and content of these directories, refer to the *IBS Data Management Guide*.

Table 4.1. IBS Directories and Protections

Directories and Files	Assigned Protections
Directories: [ONSITE], [OFFSITE], [INFOMANAGER], and [ISYSFILES]	S:RWED, O:RWED, G:RWED, W:RWED
Directories: [PNLDEV], [*Map*], and [TRNG]	S:RWED, O:RWED, G:RE, W:RE
Directory: *map*DB:[*Map*]LOG.DIR	S:RWED, O:RWED, G:RWE, W:RWE
Files in [ONSITE], [OFFSITE], [INFOMANAGER] *all files*	S:RWED, O:RWED, G:RWE, W:RWE
Files in [PNLDEV], [ISYSFILES], [*Map*], and [TRNG] *all files except those listed below*	S:RWED, O:RWED, G:RE, W:RE
MAP$MEMLOG:MEM_LOG.DAT IEMIS$EXE:SPOCC_SHARED_MEMORY.EXE	S:RWED, O:RWED, G:RWE, W:RWE
IEMIS$SYSFILES: ALLSITE.DAT MASTER_DIRECTORY.BIN OFF_TO_ON.DAT PERMITI.DAT	S:RWED, O:RWED, G:RWED, W:RWED
IEMIS$SYSFILES: OTHER_SYSTEMS.DAT	S:RWED, O:RWED, G, W

Table 4.2. PNLDEV Subdirectories

Directory Name	Contents of the Directory
[.CHAT]	User communication package
[.DATAENTR]	Dataentry forms package
[.DIGCART]	Map display, edit, and utilities software
[.DOCUMENTOR]	Tools for maintaining software documentation
[.EESF]	Emergency Exercise Simulation Facility
[.EXE]	Executable images and command files
[.FGEF]	Utilities for FGEF data conversion
[.GEOPLEX]	Utilities for CACI data conversion
[.IEMIS]	IEMIS menu program
[.IMEASY]	Economic model
[.INC]	Include files
[.KERMIT]	KERMIT file transfer software
[.LIB]	Object libraries and some commonly used source code
[.MODELS]	Models (D2, IDYNEV, CHEMS, MESORAD, OSPM)
[.MYTAPE]	MYTAPE foreign tape processing software
[.NEWSITE]	Create a new site
[.PNLDOC]	Documentation for IBS
[.REA]	Regional Evacuation Analysis software
[.RELEASE]	Command files for creating release tapes
[.REPSS]	REPSS software based on the INGRES relational database
[.SPOCC]	IBS software
[.SYS_DATA]	Forms and other run-time-fixed data
[.VAXWHO]	PNL version of the WHO software

4.2 IBS Command Files

To simplify installation and start up, command files have been prepared to support the installation and startup of IBS. These command files are necessary for building the IBS software and for enabling execution of the IBS system. Table 4.3 lists these command files and where they are called from.

Table 4.3. Command Files that Support Installation/Startup

Command File	Called From
[ISYSFILES]PNLLOGDEF.COM	SYS$MANAGER:SYSTARTUP_V5.COM defines system-wide logical names for the IBS software
[ISYSFILES]PNLEXECUT.COM	SYS$MANAGER:SYSTARTUP_V5.COM installs IBS images and executes any monitoring tasks such as weather, etc.
[ISYSFILES]PNLSYMDEF.COM	SYS$MANAGER:SYLOGIN.COM defines global symbols needed for the execution of the IBS software
[PNLDEV]LOGIN.COM	SYS$LOGIN:LOGIN.COM (1) defines symbols and logical names necessary to build the IBS software and must be executed before invoking the IBS build procedures.

(1) Only invoke [PNLDEV]LOGIN.COM from a user's SYS$LOGIN:LOGIN.COM if the user intends to modify IBS software.

4.3 IBS Software Standards

After the original installation and creation of all principal directories, a new software version may be installed by replacing only the [PNLDEV] directory tree[a]. This section describes the standards in defining subdirectories, logical names, and symbols that must be maintained if simple software replacement for upgrades is to work. These standards simplify the consistent installation and updating of IBS.

(a) It may also be necessary to install a new version of [ISYSFILES]. Refer to the installation notes accompanying the new software release.

4.3.1 Subdirectories for Software Subsystems

All IBS software for a given subsystem is contained in one subdirectory under the PNLDEV root directory. Command files named BUILDLIB.COM and BUILDEXE.COM in the subsystem's top subdirectory will build all executable software from source code for that subsystem and deposit them in IEMIS$EXE. Any logical names or symbols necessary for these procedures must be defined in DEVELOP.COM, which is to be referenced from [PNLDEV]LOGIN.COM. For example, the files for "*mysystem*" would be:

- [PNLDEV*.mysystem*]BUILDLIB.COM
- [PNLDEV*.mysystem*]BUILDEXE.COM
- [PNLDEV*.mysystem*]DEVELOP.COM.

Thus, each subsystem is allocated one line in each of [PNLDEV]BUILDLIB.COM, [PNLDEV]BUILDEXE.COM, and [PNLDEV]LOGIN.COM for the rebuild procedures. The [PNLDEV]BUILDLIB.COM and [PNLDEV]BUILDEXE.COM command files are referenced from [PNLDEV]BUILDALL.BIS, the batch program.

4.3.2 Logical Names

Logical names that are necessary for the *execution* of a subsystem must be defined in a command file located in [ISYSFILES]. This command file will be invoked from [ISYSFILES]PNLLOGDEF.COM during bootup.

Logical names required by IBS software *developers* must be assigned in a DEVELOP.COM and called from [PNLDEV]LOGIN.COM.

Appendix A contains a list of logical names used within IBS.

4.3.3 Symbols

Symbols that are necessary for the *execution* of a subsystem are to be defined in a command file located in [ISYSFILES]. This command file will be invoked from [ISYSFILES]PNLSYMDEF.COM during login.

Symbols required by IBS software *developers* must be defined in a DEVELOP.COM and called from [PNLDEV]LOGIN.COM.

Appendix A contains a list of symbol names used within IBS.

Section 5
How to Get Help

General questions regarding IBS should be directed to the FEMA IBS Project Manager:

Federal Emergency Management Agency
Attn: Ed Corvi
500 C Street SW
Washington, DC 20472
Phone (202) 646-2813

Questions regarding this installation guide or the actual installation of IBS should be directed to the IBS System Manager. The FEMA IBS System Manager can be contacted at the following address:

Federal Emergency Management Agency
Attn: Ron Wynn
POB 129
Berryville, VA 22611
Phone (703) 443-2246

All references to IBS should be made through FEMA. If an emergency contact is necessary, and FEMA cannot be reached, a second source of information is the software developer, Pacific Northwest Laboratory:

Pacific Northwest Laboratory
Attn: Blanche M. Bailey, K7-22
POB 999, MS K7-22
Richland, WA 99352
Phone (509) 375-2615

Appendix A
Logical Names and Symbols

The system logical names and symbol definitions required by IBS users are defined in command files located in the PNLDEVDB:[ISYSFILES] directory.

- PNLDEVDB:[ISYSFILES]PNLLOGDEF.COM executes several other command files that assign system-level logical names. This occurs at boot time only: PNLLOGDEF.COM is called from SYS$MANAGER:SYSTARTUP_V5.COM.

ASS_DISK_LOG.COM	-- Disk logical names for IBS. This file will need to be modified for each new installation and saved before re-installing a new release of the software.
ASS_IBS_LOG.COM	-- Various logical names required for all parts of IBS.
ASS_MAP_LOG.COM	-- Logical names which point to the map data.
ASS_SITE_LOG.COM	-- Logical names which point to site-specific data.
ASS_TRNG_LOG.COM	-- Logical names for training functions
ASS_VENDOR_LOG.COM	-- Logical names for vendor-supplied software.[a]

- PNLDEVDB:[ISYSFILES]PNLSYMDEF.COM executes several other command files that define symbol names. This occurs at login time for each user: PNLSYMDEF.COM is called from SYS$MANAGER:SYLOGIN.COM.

ASS_IBS_SYM.COM	-- Symbol definitions for IBS in general.
ASS_MAP_SYM.COM	-- Symbol definitions for switching between map databases.
ASS_VENDOR_SYM.COM	-- Symbol definitions for vendor-supplied software.[b]

Any special logical names or symbol names required by IBS software developers will be assigned or defined in PNLDEVDB:[PNLDEV]LOGIN.COM (or in a DEVELOP.COM file called by it).

One other IBS logical name, PNLDEVDB, which points to the disk device where the IBS software ([PNLDEV] and [ISYSFILES]) resides, needs to be defined. It

---

(a) The Intelligent Machine Model (TIMM) is but one of the possible vendor-supplied software packages.

(b) Weather and Presentation Graphics are but two of the possible vendor-supplied software packages.

must be in SYS$MANAGER:SYSTARTUP_V5.COM immediately before executing PNLDEVDB:[ISYSFILES]PNLLOGDEF.COM.

The IBS system-level logical names and symbols in this appendix are organized in the following five lists:

- Disk logical names
- IBS logical names and symbols
- *Map* logical names and symbols
- *Site* logical names
- Other logical names and symbols.

A.1 Disk Logical Names

Disk logicals (Change at installation time)

Value	Logical Name	
DUA1:	EVNDB	! InfoManager
DUA1:	TRNGDB	! Training
DUA1:	***Map***DB	! Map database

A.2 IBS Logical Names and Symbols

Logicals for IBS

Value	Logical Name
PNLDEVDB:[PNLDEV.EXE]	IEMIS$EXE:
PNLDEVDB:[PNLDEV.INC]	IEMIS$INC:
PNLDEVDB:[PNLDEV.IEMIS.DATAENTRY]	IEMIS$FRM:
PNLDEVDB:[PNLDEV.SYS_DATA]	PNL$SYS_DATA
PNLDEVDB:[PNLDEV.SYS_DATA]	PNL$HELP:
PNLDEVDB:[PNLDEV.SYS_DATA.MSG]	IEMIS$MSG
PNLDEVDB:[ISYSFILES]	IEMIS$SYSFILES
IEMIS$EXE:	MAP$SYSTEM
PNL$SYS_DATA:IEMIS_MAP_HELP.DAT	MAP$HELP$RAW
PNL$SYS_DATA:IEMIS_MAP_HELP.BIN	MAP$HELP
IEMIS$SYSFILES:BLANKICON.DMS	BLANKICON
IEMIS$SYSFILES:COLORTBL.DAT	MAP$COLOR$TABLE
IEMIS$SYSFILES:PROBLEM.RPT	PROBLEM
PNL$HELP,SYS$SYSROOT:[SYSHLP]	SYS$HELP
IEMIS$EXE:SPOCC_SHARED_MEMORY.EXE	SPOCC_SHARED_MEMORY

Logicals for REPSS

Value	Logical Name
IEMIS$SYSFILES:,IEMIS$EXE:	PNL$SYSFILES

Logicals for TRPLOT (Digitizing)

Value	Logical Name
SUMMA	TRPLOT$DIGITIZER
DIGITIZER	SUMMA

Logicals for DataEntry

Value	Logical Name
PNLDEVDB:[PNLDEV.DATAENTR.F77]	DATAENTRY_DIR:
PNLDEVDB:[PNLDEV.DATAENTR.F77.TBLGEN]	TBL_GEN:

Logicals for the Chems Model

Value	Logical Name
PNLDEVDB:[PNLDEV.SYS_DATA.MHACS_DATA]	PROPDAT

Logicals for the Mesorad Model

Value	Logical Name
PNLDEVDB:[PNLDEV.SYS_DATA.LABELS]	MES_LABELS

Command files

Symbol	Symbol Definition
ALLSITE	:== @IEMIS$SYSFILES:ALLSITE
FTPSTART	:== @IEMIS$SYSFILES:FTPSTART
FTPSTOP	:== @IEMIS$SYSFILES:FTPSTOP
SELECTDB	:== @IEMIS$SYSFILES:SELECTDB
DELUSER	:== @IEMIS$EXE:DELUSER
DOLIMITXX	:== @IEMIS$EXE:DOLIMITXX
DOSEELIMITXX	:== @IEMIS$EXE:DOSEELIMITXX
FDISP	:== @IEMIS$EXE:FDISP
FIXMAPDB	:== @IEMIS$EXE:FIXMAPDB
FIXSITEDB	:== @IEMIS$EXE:FIXSITEDB
FPRINT	:== @IEMIS$EXE:FPRINT
IEMIS	:== @IEMIS$EXE:IEMIS
LIMITXX	:== @IEMIS$EXE:LIMITXX
LIMITXX2	:== @IEMIS$EXE:LIMITXX2
NEWUSER	:== @IEMIS$EXE:NEWUSER

Foreign commands

Symbol	Symbol Definition
KERMIT	:== $ IEMIS$EXE:KERMIT
MR_CHK	:== $ IEMIS$EXE:XMAIL CHECK_MAIL
SEELIMITXX	:== $ IEMIS$EXE:SEELIMITXX
XMAIL	:== $ IEMIS$EXE:XMAIL

IBS Utilities

Symbol	Symbol Definition
CSEPP	:== RUN/NODEB IEMIS$EXE:CSEPP
CSEPP_NOGR	:== RUN/NODEB IEMIS$EXE:CSEPP_NOGR
DELSITE	:== RUN/NODEB IEMIS$EXE:DELSITE
EVENT	:== @IEMIS$EXE:EVENT.BIS

IBS Utilities *(continued)*

Symbol	Symbol Definition
IBS	:= = RUN/NODEB IEMIS$EXE:CSEPP
IBS_NOGR	:= = RUN/NODEB IEMIS$EXE:CSEPP_NOGR
LOCATE	:= = RUN/NODEB IEMIS$EXE:LOCATE
MAIL_C*HECKER	.= = @IEMIS$EXE:MAIL_C.BIS
MAIL_LIST	:= = RUN/NODEB IEMIS$EXE:MAIL_LIST
MBMGR	:= = RUN/NODEB IEMIS$EXE:MBMGR
MLMGR	:= = RUN/NODEB IEMIS$EXE:MLMGR
ONPOSTSIM	:= = RUN/NODEB IEMIS$EXE:ONPOSTSIM
PDMGR	:= = RUN/NODEB IEMIS$EXE:PDMGR
RAMGR	:= = RUN/NODEB IEMIS$EXE:RAMGR
RMMGR	:= = RUN/NODEB IEMIS$EXE:RMMGR
SITEBKP	:= = RUN/NODEB IEMIS$EXE:SITEBKP

IEMIS Utilities

Symbol	Symbol Definition
ADDATTDMS	:= = RUN/NODEB IEMIS$EXE:ADDATTDMS
APPENDDMS	:= = RUN/NODEB IEMIS$EXE:APPENDDMS
ASCIIXDMS	:= = RUN/NODEB IEMIS$EXE:ASCIIXDMS
ASCXATT	:= = RUN/NODEB IEMIS$EXE:ASCXATT
ATTXASC	:= = RUN/NODEB IEMIS$EXE:ATTXASC
CLIPDMS	:= = RUN/NODEB IEMIS$EXE:CLIPDMS
COLORDMS	:= = RUN/NODEB IEMIS$EXE:COLORDMS
CREATELIMITS	:= = RUN/NODEB IEMIS$EXE:CREATLIMITS
CREATETAC	:= = RUN/NODEB IEMIS$EXE:CREATETAC
CREATLIMITS	:= = RUN/NODEB IEMIS$EXE:CREATLIMITS
DECLUTTER_TOPO	:= = RUN/NODEB IEMIS$EXE:DECLUTTER_TOPO
DLGXDMS	:= = RUN/NODEB IEMIS$EXE:DLGXDMS
DMSXASCII	:= = RUN/NODEB IEMIS$EXE:READDMS
DMSXDLG	:= = RUN/NODEB IEMIS$EXE:DMSXDLG
DMSXFGEF	:= = RUN/NODEB IEMIS$EXE:DMSXFGEF
DMSXPMF	:= = RUN/NODEB IEMIS$EXE:DMSXPMF
EXTRACT	:= = RUN/NODEB IEMIS$EXE:EXTRACT
FGEFXDMS	:= = RUN/NODEB IEMIS$EXE:FGEFXDMS
FLOPYXDMS	:= = RUN/NODEB IEMIS$EXE:FLOPYXDMS
GEDIT	:= = RUN/NODEB IEMIS$EXE:GEDIT
LATLONXUTM	:= = RUN/NODEB IEMIS$EXE:LATLONXUTM
LINEAR_JOIN	:= = RUN/NODEB IEMIS$EXE:LINEAR_JOIN
MAKEDMS	:= = RUN/NODEB IEMIS$EXE:MAKEDMS
MAPDBGEN	:= = RUN/NODEB IEMIS$EXE:MAPDBGEN
MAPDBUPDATE	:= = RUN/NODEB IEMIS$EXE:MAPDBUPDT
MAPDBUPDT	:= = RUN/NODEB IEMIS$EXE:MAPDBUPDT
MODATTDMS	:= = RUN/NODEB IEMIS$EXE:MODATTDMS
MODTEXTNM	:= = RUN/NODEB IEMIS$EXE:MODTEXTNM
MPDISPLAY	:= = RUN/NODEB IEMIS$EXE:MPDISPLAY
NEWSITE	:= = RUN/NODEB IEMIS$EXE:NEWSITE
PLOTDMS	:= = RUN/NODEB IEMIS$EXE:PLOTDMS
PMFXDMS	:= = RUN/NODEB IEMIS$EXE:PMFXDMS

IEMIS Utilities *(continued)*

Symbol	Symbol Definition
POPXDMS	:= = RUN/NODEB IEMIS$EXE:POPXDMS
READDMS	:= = RUN/NODEB IEMIS$EXE:READDMS
SEELIMITS	:= = RUN/NODEB IEMIS$EXE:SEELIMITS
SHOWATT	:= = RUN/NODEB IEMIS$EXE:SHOWATT
SMOOTHER	:= = RUN/NODEB IEMIS$EXE:SMOOTHER
STATDMS	:= = RUN/NODEB IEMIS$EXE:STATDMS
TAC3XCONTXDMS	:= = RUN/NODEB IEMIS$EXE:TAC3XCONTXDMS
TBXDMS	:= = RUN/NODEB IEMIS$EXE:TBXDMS
TESTICON*S	:= = RUN/NODEB IEMIS$EXE:TESTICON
TIGERCVT	:= = RUN/NODEB IEMIS$EXE:TIGERCVT
TIGERXDMS	:= = RUN/NODEB IEMIS$EXE:TIGERXDMS
UPDATEATT	:= = RUN/NODEB IEMIS$EXE:UPDATEATT
USERDBGEN	:= = RUN/NODEB IEMIS$EXE:USERDBGEN[(a)]
USGS100KXDMS	:= = RUN/NODEB IEMIS$EXE:USGS100KXDMS
USGSXDMS	:= = RUN/NODEB IEMIS$EXE:USGSXDMS
USGSXTAC3	:= = RUN/NODEB IEMIS$EXE:USGSXTAC3

Miscellaneous

Symbol	Symbol Definition
OSPM_GR	:= = RUN/NODEB IEMIS$EXE:OSPM_GR
OSPM_NOGR	:= = RUN/NODEB IEMIS$EXE:OSPM_NOGR
SAY	:= = WRITE SYS$OUTPUT
SETUPTEK	:= = RUN/NODEB IEMIS$EXE:SETUPTEK
TEK	:= = ASSIGN/JOB TEKTRONIX TRPLOT$DEVICE

A.3 *Map* Logical Names and Symbols

Default map database (Change at installation time)

Value	Logical Name
MapDB:[***Map***]	MAP$DATA$MAIN
MapDB:[***Map***.]	MAP$TOPDIR

Directories and Control Files

Value	Logical Name
MAP$TOPDIR:[ATT]	MAP$ATTRIB
MAP$TOPDIR:[ICON]	MAP$ICONS
MAP$TOPDIR:[LOG]	MAP$MEMLOG
MAP$ICONS:ICON.CTL	MAP$ICON$CTL
MAP$DATA$MAIN:TOPOCTL.FIL	MAP$CONTROL
MAP$DATA$MAIN:TOPODESC.FIL	MAP$TOPOGRAPHY
MAP$ATTRIB:ATTRIBDIC.BIN	MAP$ATTRIBUTE$DICTIONARY

(a) The User Database Generation utility (USERDBGEN) is not currently supported, though it is still documented as it once was.

Topographies

Value	Logical Name
MAP$TOPDIR:[xx]	MAP$DATA$xx
MAP$DATA$xx	T$xx

Support for Old Map Databases

Value	Logical Name
MAP$DATA$xx	xx$TXT[a]
MAP$DATA$GN	MAP$GNIS
MAP$DATA$PL	MAP$PPL

Symbols to switch between map databases for individual users (Change at installation time)

Symbol	Symbol Definition
*Map*DATA*BASE	:= = @*Map*DB:[*Map*]SETUPDB JOB

Symbols to switch between map databases for all users (Change at installation time)

Symbol	Symbol Definition
SYS*Map*DATA*BASE	:= = @*Map*DB:[*Map*]SETUPDB SYSTEM

A.4 *Site* Logical Names

Default Operational Site Database (Change at installation time)

Value	Logical Name
EVNDB:	EVNT$DISK
INFOMANAGER	EVNT$USER
site	EVNT$SITE

Offpost directories

Value	Logical Name
EVNDB:[OFFSITE]	OFFSITE$DIR
EVNDB:[OFFSITE.*site*]	OFF$DIR
EVNDB:[OFFSITE.*site*.]	OFF$MAPDIR

Onpost directory

Value	Logical Name
EVNDB:[ONSITE]	ONSITE$DIR

InfoManager's home directory

Value	Logical Name
EVNDB:[INFOMANAGER]	INFO$DIR

(a) These logicals are maintained merely to support old versions of map databases. It is strongly recommended that DMS files using these logicals to refer to TXT files be corrected to use the T$xx logicals.

Set site directory logicals

Value	Logical Name
EVNDB:[INFOMANAGER.SITES.*site*]	SITE$DIR
EVNDB:[INFOMANAGER.SITES.*site*.]	SITE$TOPDIR
EVNDB:[INFOMANAGER.SITES.*site*.INPUT.MAPS.]	SITE$MAPDIR

Miscellaneous

Value	Logical Name
SITE$TOPDIR:[BCK]	SITE$BCK
SITE$TOPDIR:[INPUT.IP]	SITE$IP
SITE$TOPDIR:[INPUT.OFF]	SITE$OFF
SITE$TOPDIR:[INPUT.POP]	SITE$POP

Models

Value	Logical Name
SITE$TOPDIR:[INPUT.CHEMS]	SITEICHEMS
SITE$TOPDIR:[INPUT.D2]	SITEID2
SITE$TOPDIR:[INPUT.DYNEV]	SITEIDYNEV
SITE$TOPDIR:[INPUT.MESORAD]	SITEIMESORAD
SITE$TOPDIR:[INPUT.OSPM]	SITEIOSPM
SITE$TOPDIR:[OUTPUT.CHEMS]	SITEOCHEMS
SITE$TOPDIR:[OUTPUT.D2]	SITEOD2
SITE$TOPDIR:[OUTPUT.DYNEV]	SITEODYNEV
SITE$TOPDIR:[OUTPUT.MESORAD]	SITEOMESORAD
SITE$TOPDIR:[OUTPUT.OSPM]	SITEOOSPM

Volatile topographies

Value	Logical Name
SITE$MAPDIR:[D2]	MAP$DATA$D2
SITE$MAPDIR:[EX]	MAP$DATA$EX
SITE$MAPDIR:[FI]	MAP$DATA$FI
SITE$MAPDIR:[VA]	MAP$DATA$VA
OFF$MAPDIR:[DA]	MAP$DATA$DA
OFF$MAPDIR:[EI]	MAP$DATA$EI
OFF$MAPDIR:[RM]	MAP$DATA$RM
OFF$MAPDIR:[RP]	MAP$DATA$RP
OFF$MAPDIR:[SR]	MAP$DATA$SR
OFF$MAPDIR:[WD]	MAP$DATA$WD
OFF$MAPDIR:[WI]	MAP$DATA$WI
MAP$DATA$EI	T$EI

A.5 Other Logical Names and Symbols

External software packages (To be changed at installation time)

Value	Logical Name
DISK1:[TIMM]	SYS$TIMM[a]

Logicals for Training

Value	Logical Name
TRNGDB:[TRNG.TXT]	TRAINING$TXT
TRNGDB:[TRNG.EXER]	TRAINING$EXER

External software packages (To be changed at installation time)

Symbol	Symbol Definition
DP	:== TYPE/PAGE IEMIS$EXE:NOPRGRAF.MSG[b]
WEATHER	:== TYPE/PAGE IEMIS$EXE:NOWEATHER.MSG[c]

(a) If The Intelligent Machine Model software package exists on the system, ensure that the SYS$TIMM logical points to the software.

(b) If the Presentation Graphics software package exists on the system, ensure that the DP symbol runs the software.

(c) If the Weather Access software package exists on the system, ensure that the WEATHER symbol runs the software.

Appendix B
Generation of IBS Distribution Tapes

The IBS is distributed on tape. The installation tapes are to be in standard VMS Backup format with separate save sets for each main IBS directory. The current directories and their save set names are listed in Table B.1.

Table B.1. List of Save Sets and Procedures for Generating Tapes

Directory	Save Set	Tape	Procedure
ISYSFILES	ISYSFILES.BCK	1	RELEASE_PNLDEVDB.COM
PNLDEV	PNLDEV.BCK	1-2(1)	RELEASE_PNLDEVDB.COM
OFFSITE	OFFSITE.BCK	3	RELEASE_EVNDB.COM
ONSITE	ONSITE.BCK	3	RELEASE_EVNDB.COM
INFOMANAGER	INFOMANAGER.BCK	3	RELEASE_EVNDB.COM
TRNG	TRNG.BCK	4	RELEASE_TRNGDB.COM
Map	*Map*.BCK	5-?	RELEASE_*Map*DB.COM

(1) If you are using TK70 tapes, the PNLDEV.BCK save set may fit on a single tape.

B.1 Preparing for an IBS Release

The following items summarize an approach to preparing a releasable copy of the IBS BEFORE generating the IBS distribution tapes:

1. Backup the system to another disk device, including these full subdirectories:

   - [PNLDEV...]
   - [ISYSFILES...]
   - [OFFSITE...]
   - [ONSITE...]
   - [INFOMANAGER...]

2. Reassign the logical names PNLDEVDB and EVNDB to point to the new disk location.

3. Rebuild the IBS on the new disk (following the procedures outlined in Appendix D).

4. From the [OFFSITE] and [INFOMANAGER] directories, remove all sites that are not applicable to the user for whom you are preparing the distribution tapes.

5. Update [ISYSFILES] files for the remaining sites with correct values for longitude-latitude or Universal Transverse Mercator (UTM) location.

   These files include

   - C000CHEMS.GEN
   - C000D2INP.DAT

   and others.

6. Update the ONPOSTSIM files to reflect the site location. (Refer to the *IBS Data Management Guide* for a list of these files.)

7. Clean up, remove, or otherwise police all files that do NOT belong in the IBS directories. (Refer to the *IBS Data Management Guide* for a complete list of IBS directories and files.)

8. Create an empty MAP$MEMLOG:MEM_LOG.DAT file.

B.2 Preparing the Distribution Tapes

To create distribution tapes of the prepared system:

1. Obtain at least six[(a)] TK50 or TK70 tapes which are either unused or contain nothing of value.

2. Assign the TAPE logical to point to the correct tape backup device:
   $ ASSIGN *TapeA*: TAPE

3. Initialize the tapes using the VMS utility INIT:[(b)]
   $ INIT TAPE: EVNDB [or **PNLDEV**, **TRNG**, ***Map***]

4. Warn all users to stop using the IBS software and databases:

(a) The number required depends on the type of tape and the size of the map database, which varies between 60 MB and 2 GB.

(b) If using option 5.A, then you need to initialize all but the first tape for each of the PNLDEVDB and *Map*DB backups.

```
$ CHAT/ALL
**********************************************************************

                ATTENTION ALL IBS USERS

          IBS Backup about to commence.  Logout now.

**********************************************************************
<CTRL-Z>
```

5. Create the installation tapes by A) using the command files and the procedure provided here for that purpose or B) using the manual procedure shown here. Repeat the chosen procedure (A or B below) for each save set.

   Option A: Use the command files in PNLDEVDB:[PNLDEV.RELEASE].

   $ SET DEFAULT PNLDEVDB:[PNLDEV.RELEASE]
   $ @RELEASE_LOG ***procedure Map*** (See the table on page B.1.)
   Y (Yes, you've initialized the tapes.)
   Destination (to whom you're releasing the tapes)
   Comment 1
   Comment 2
   ...
   Comment n
   <RETURN>

   The backup will then proceed.

   For example:

   $ SET DEFAULT PNLDEVDB:[PNLDEV.RELEASE]
   $ @RELEASE_LOG RELEASE_MAPDB.COM UMDA
   Y
   Salem, OR
   04/20/92, JRW. Beta release;
   <RETURN>

   Option B: Do everything manually.

   a. Mount the tape FOREIGN using the VMS utility MOUNT:
      $ MOUNT/FOREIGN TAPE:

   b. De-install the shared image file. If you do not, the file will not be backed up and the system on which the tapes are to be installed will lack that one file, which is so important that many of the IBS application will NOT run. A complete rebuild of IBS would be necessary.

      $ @PNLDEVDB:[PNLDEV.RELEASE]INSTALL-OFF

c. Create empty ALLSITE, OTHER_SYSTEMS, and PERMITI files:

```
$ SET DEFAULT PNLDEVDB:[ISYSFILES]
$ COPY ALLSITE.DAT .BKP
$ CREATE ALLSITE.DAT                <CTRL-Z>
$ COPY OTHER_SYSTEMS.DAT .BKP
$ CREATE OTHER_SYSTEMS.DAT    <CTRL-Z>
$ COPY PERMITI.DAT .BKP
$ SEAR PERMITI.BKP "NOT - A NAME"/OUT=PERMITI.DAT
```

d. BACKUP the necessary files to tape.

Tapes 1 and 2:

```
$ BACKUP /IGNORE=LABEL PNLDEVDB:[ISYSFILES]*.*
     TAPE:ISYSFILES.BCK /EXC=.BKP
$ BACKUP /IGNORE=LABEL PNLDEVDB:[PNLDEV..]*.*  TAPE:PNLDEV.BCK
```

Tape 3:

```
$ BACKUP /IGNORE=LABEL EVNDB:[OFFSITE...]*.*  TAPE:OFFSITE.BCK
$ BACKUP /IGNORE=LABEL EVNDB:[ONSITE...]*.*  TAPE:ONSITE.BCK(a)
$ BACKUP /IGNORE=LABEL EVNDB:[INFOMANAGER...]*.* TAPE:INFOMANAGER.BCK
```

Tape 4:

```
$ BACKUP /IGNORE=LABEL TRNGDB:[TRNG...]*.*  TAPE:TRNG.BCK
```

Tapes 5-?:

```
$ BACKUP /IGNORE=LABEL MapDB:[Map...]*.*  TAPE:Map.Bck
```

e. Restore the ALLSITE, OTHER_SYSTEMS, and PERMITI files:

```
$ SET DEFAULT PNLDEVDB:[ISYSFILES]
$ REN ALLSITE.BKP .DAT
$ REN OTHER_SYSTEMS.BKP .DAT
$ REN PERMITI.BKP .DAT
```

f. Re-install shared the image file:

```
$ @PNLDEVDB:[PNLDEV.RELEASE]INSTALL-ON
```

If you do not do this, many of the IBS applications will not run.

g. Dismount the tape:

```
$ DISMOUNT TAPE:
```

6. On each tape label, indicate the date, save set names, and storage requirements.

(a) The ONSITE directory should be empty except for JUNK.TMP:

```
$ DEL EVNDB:[ONSITE]*.*;* /EXCLUDE=JUNK.TMP
```

7. Set the write protect switch on each tape.

APPENDIX C
Disk Requirements for Standard Topographies

It is often necessary to estimate the amount of disk space required for a given application. One significant use of disk space in any IBS application is the storage of map data sets (topographic data) for the IBS Geographic Information System (GIS). The amount of disk space required to store GIS data on disk is proportional to the number of graphic objects in the data and their complexity. The following table provides "worst-case" (high-range) estimates of the quantity of data in different topographic overlays for 1:2,000,000-scale data from the U.S. Geological Survey (USGS) within a longitude-latitude range of about 1 degree by 1 degree. This information may be useful in estimating the storage requirements of commonly used data from the 1:2,000,000-scale USGS data sets.

Certain limits imposed on the software are based on parameters involving the number of objects. You can use the IBS utility STATDMS to examine specific DMS topographic data files and determine the number of objects in those files. If you then relate the block sizes of the examined files to the "worst-case" estimates in the table, you can roughly estimate 1) the quantity of detail for typical topographic data, and 2) the area (data sets) that a given system will be able to load into memory.

Table C.1. Worst-Case Block Size Estimates for IBS Topographic Data (1:2,000,000-scale USGS, 1x1 degrees Long-Lat)

DMS Topography	Blocks	DMS Topography	Blocks
Admin. Boundaries	1096	HAZMAT Sites	592
Congressional Dist.	718	NAPB Targets	112
Cultural Features	64	Nuclear Power Plants	16
Electric Power Grid	364	Political Boundaries	1734
Color Flood Points	--	Population (1980 Census)	842
Features (Icons)	--	Population Place Names	2896
GNIS Geog. Names	15024	Railroads	2144
DMS data	*6496*	Roads & Trails	2530
Text	*8528*	Streams	7232
Hypsography	112	Water Bodies	3668

Appendix D
Rebuilding IBS from the Source Code

The following procedure will rebuild the entire IBS system. This procedure can be expected to require several hours of machine time and 50,000 blocks of scratch disk space on most VMS systems: It should NOT be run while IBS users are logged on to the system.

1. Verify that the source code does exist on the system:
   $ DIRECTORY PNLDEVDB:[PNLDEV...]

2. Verify that the system has a FORTRAN compiler and linker:
   $ CREATE 1.FOR
   PRINT *,'HELLO'
   END
   <CTRL-Z>
   $ FORTRAN 1
   $ LINK 1
   $ RUN 1

   If there are any compilation or link errors, or if the result of the RUN command is not the line 'HELLO', then you will not be able to rebuild.

3. Make certain you have both SYSPRV and CMKRNL privilege:
   $ SET PROC/PRIV=(SYSPRV,CMKRNL)

   If there is an error, you will not be able to rebuild.

4. Start the batch job to do the rebuild:
   $ SUBMIT/NOPRINT/NOTIFY/KEEP PNLDEVDB:[PNLDEV]BUILDALL.BIS

5. Periodically look at the log file to determine if there are any problems:
   $ TYPE SYS$LOGIN:BUILDALL.LOG

6. When the rebuild has completed, you may want to clean up some of the documentation files. These are sometimes useful to developers but mostly just take up space (35 MB), being complete copies of the source code at the time of the past two rebuilds.

   $ DELETE PNLDEVDB:[PNLDEV.PNLDOC]*X.X*.*

PNL-8282
UC-620

DISTRIBUTION

No. of Copies

OFFSITE

2 DOE/Office of Scientific and Technical Information

5 Federal Emergency Management Agency
Attn: Ed Corvi
500 C Street SW
Washington, DC 20472

Federal Emergency Management Agency
Attn: Ralph Sinnot
500 C St. SW
Washington, D.C. 20472

Madhu Beriwal
Innovative Emergency Management
2351 Energy Drive, Suite 3009
Baton Rouge, LA 70808

COL Lawrence E. Skelly
United States Army Nuclear and Chemical Agency
7500 Backlick Road, Building 5073
Springfield, VA 22150-3198

John Bower
Bower Software Services
104 E. 36th Place
Kennewick, WA 99337

No. of Copies

ONSITE

1 DOE Richland Field Office

R. B. Goranson

26 Pacific Northwest Laboratory

B. M. Bailey (10)
M. J. Burford
T. R. Downing
S. W. Matsumoto (3)
W. D. Millard
S. L. Savelli
E. E. Schrank
J. R. Williams
C. Winters (3)
Publishing Coordination
Technical Report Files (3)

DATE
FILMED
4/20/94
END